

**IN THE UNITED STATES DISTRICT COURT  
FOR THE DISTRICT OF DELAWARE**

INTERNATIONAL BUSINESS MACHINES CORPORATION,	)	
	)	
	)	
Plaintiff,	)	
	)	
v.	)	C.A. No. 15-137-LPS
	)	
THE PRICELINE GROUP INC., KAYAK SOFTWARE CORPORATION, OPENTABLE, INC., AND PRICELINE.COM LLC	)	
	)	<b>JURY TRIAL DEMANDED</b>
	)	
Defendants.	)	

**DEFENDANTS' OPENING CLAIM CONSTRUCTION BRIEF**

Francis DiGiovanni (#3189)  
Thatcher A. Rahmeier (#5222)  
DRINKER BIDDLE & REATH LLP  
222 Delaware Avenue, Suite 1410  
Wilmington, DE 19801  
(302) 467-4200  
francis.digiovanni@dbr.com  
thatcher.rahmeier@dbr.com

Dan D. Davison  
NORTON ROSE FULBRIGHT US LLP  
2200 Ross Ave., Suite 3600  
Dallas, TX 75201  
(214) 855-8000  
dan.davison@nortonrosefulbright.com

Dated: July 26, 2016

Richard S. Zembek  
Daniel S. Leventhal  
Daniel A. Prati  
NORTON ROSE FULBRIGHT US LLP  
Fulbright Tower  
1301 McKinney, Suite 5100  
Houston, TX 77010  
(713) 651-5283  
richard.zembek@nortonrosefulbright.com  
daniel.leventhal@nortonrosefulbright.com  
dan.prati@nortonrosefulbright.com

Gilbert A. Greene  
Marwan Elrakabawy  
NORTON ROSE FULBRIGHT US LLP  
98 San Jacinto Boulevard, Suite 1100  
Austin, TX 78701  
(512) 474-5201  
bert.greene@nortonrosefulbright.com  
marwan.elrakabawy@nortonrosefulbright.com

*Attorneys for Defendants The Priceline  
Group Inc., Priceline.com LLC, KAYAK  
Software Corporation, and OpenTable, Inc.*

**TABLE OF CONTENTS**

	<b>Page</b>
I. INTRODUCTION .....	1
II. DISPUTED TERMS COMMON TO THE ‘967 AND ‘849 PATENTS .....	1
A. Overview of the ‘967 and ‘849 Patents.....	1
B. objects .....	2
C. application(s).....	4
D. computer network / network .....	8
III. DISPUTED TERMS FROM THE ‘967 PATENT .....	11
A. the objects being retrieved from the objects stored at the respective reception system, or if unavailable from the objects stored at the respective reception system, then from the network.....	11
B. the method of claim 2 wherein the predetermined plan.....	12
C. permit random movement .....	14
D. at least one procedure for navigating to a new application / a plurality of different procedure [sic] for navigating to a new application.....	15
E. storage control parameter.....	17
IV. DISPUTED TERMS FROM THE ‘849 PATENT .....	17
A. storing a predetermined amount of the advertising data in a store established at the respective reception systems .....	17
B. selectively storing advertising objects at a store established at the reception system.....	19
C. structuring advertising in a manner compatible to that of the applications so that it may be presented.....	20
V. DISPUTED TERMS FROM THE ‘601 PATENT .....	21
A. Overview of the ‘601 Patent .....	21
B. order of the independent claim steps .....	22
C. recursively embedding the state information in all identified continuations .....	24
D. continuation.....	25
E. means-plus function terms .....	26
F. [dynamically downloading computer program code to the client to] perform said step of embedding which is responsive to said step of communicating the output to the client.....	28

**TABLE OF CONTENTS**  
(continued)

	<b>Page</b>
G.    filtering one of said hyperlinks and data output from said services according to a predetermined criteria / adding one of said hyperlinks and data to said output from said services according to a predetermined criteria .....	29
VI.    DISPUTED TERMS FROM THE ‘346 PATENT .....	30
A.    federated computing environment .....	30
VII.   CONCLUSION.....	30

**TABLE OF AUTHORITIES**

	<b>Page(s)</b>
<b>CASES</b>	
<i>Cronos Techs., LLC v. Expedia, Inc.</i> , No. 13-1538-LPS, slip op. (June 20, 2015) .....	22
<i>Dow Chem. Co. v. Nova Chemicals Corp. (Canada)</i> , 803 F.3d 620 (Fed. Cir. 2015).....	7
<i>Halliburton Energy Servs., Inc. v. M-I LLC</i> , 514 F.3d 1244 (Fed. Cir. 2008).....	4
<i>Helmsderfer v. Bobrick Washroom Equip., Inc.</i> , 527 F.3d 1379 (Fed. Cir. 2008).....	3
<i>Med. Instrumentation &amp; Diagnostics Corp. v. Elekta AB</i> , 344 F.3d 1205 (Fed. Cir. 2003).....	26, 27
<i>Merck &amp; Co. v. Teva Pharm. USA, Inc.</i> , 395 F.3d 1364 (Fed. Cir. 2005).....	17, 24, 29, 30
<i>Mformation Techs., Inc. v. Research in Motion Ltd.</i> , 764 F.3d 1392 (Fed. Cir. 2014).....	22
<i>Nautilus, Inc. v. Biosig Instr., Inc.</i> , 134 S.Ct. 2120 (2014).....	4, 7, 14, 29
<i>Novo Industries, L.P. v. Micro Molds Corp.</i> , 350 F.3d 1348 (Fed. Cir. 2003).....	13
<i>Nystrom v. TREX Co.</i> , 424 F.3d 1136 (Fed. Cir. 2005).....	8
<i>Phillips v. AWH Corp.</i> , 415 F.3d 1303 (Fed. Cir. 2005).....	8
<i>Ruckus Wireless, Inc. v. Innovative Wireless Solutions, LLC</i> , Nos. 2015-1425 & 1438, 2016 WL 3065024 (Fed. Cir. May 31, 2016) .....	10, 11
<i>TALtech Ltd. v. Esquel Apparel, Inc.</i> , 279 F. App'x. 974 (Fed. Cir. 2008) .....	22
<i>Thorner v. Sony Computer Entm't Am. LLC</i> , 669 F.3d 1362 (Fed. Cir. 2012).....	2
<i>TQ Beta LLC v. Dish Network Corp.</i> , 14-CV-848-LPS-CJB, 2016 WL 356024 (D. Del. Jan. 28, 2016) .....	18

**TABLE OF AUTHORITIES**

(Continued)

	<b>Page(s)</b>
<i>Trusted Knight Corp. v. Int’l Bus. Machines Corp.</i> , 14-cv-1063-LPS-CJB, 2015 WL 7307134 (D. Del. Nov. 19, 2015) .....	13

**OTHER AUTHORITIES**

IBM Dictionary of Computer, Tenth Edition (Aug 1993) .....	24
Webster’s New World Dictionary of Computer Terms, Eighth Edition (2000) .....	24
Webster’s Computer Dictionary (1994) .....	24
Microsoft Computer Dictionary, Fourth Edition (1999) .....	24

**TABLE OF EXHIBITS**

All exhibits attached to Joint Claim Construction Statement.

All emphasis in quotations added unless otherwise indicated.

NO.	EXHIBIT
A-1	U.S. Patent No. 5,796,967 (“the ‘967 Patent”)
A-3	July 16, 1997, Amendment in ‘967 Patent File History
A-4	October 12, 1995, Amendment in ‘967 Patent File History
A-5	October 21, 1994, Disclosure Statement in ‘967 Patent File History
A-22	July 9, 2004, Office Action in U.S. Patent Application No. 10/205,393
A-25	January 30, 2004, Office Action in U.S. Patent Application No. 09/656,581
A-27	Original Claims of the ‘967 Patent
A-28	July 16, 1997, Amendment in ‘967 Patent File History
B-1	U.S. Patent No. 7,072,849 (“the ‘849 Patent”)
B-2	February 27, 2002, Decision on Appeal in ‘849 Patent File History
B-3	December 23, 2005, Decision on Appeal in ‘849 Patent File History
B-4	October 19, 1994, Amendment in ‘849 Patent File History
B-5	November 1, 1996, Examiner Interview Summary in ‘849 Patent File History
B-8	October 1, 2004, Appeal Brief in ‘849 Patent File History
B-11	September 27, 2002, Decision on Request for Rehearing in ‘849 Patent File History
B-18	April 29, 2002, Request for Rehearing in ‘849 Patent File History
B-19	October 27, 1997, Office Action in ‘849 Patent File History
C-1	U.S. Patent No. 5,961,601 (“the ‘601 Patent”)
C-4	IBM Dictionary of Computer, Tenth Edition (Aug 1993)
C-5	Webster’s New World Dictionary of Computer Terms, Eighth Edition (2000)

**TABLE OF EXHIBITS**

(Continued)

NO.	EXHIBIT
C-6	Webster's Computer Dictionary (1994)
C-7	Microsoft Computer Dictionary, Fourth Edition (1999)
D-1	U.S. Patent No. 7,631,346 ("the '346 Patent")

## I. INTRODUCTION

Defendants’ constructions reflect the plain claim language and assign meaning to every term. IBM seeks to remove claim limits from the ‘601 Patent and improperly broaden the already broad and abstract claims of the ‘967 and ‘849 Patents to cover an architecture fundamentally incompatible with the claims. IBM’s constructions cannot be reconciled with its arguments made to avoid a Section 101 finding. For example, when it needed the ‘601 Patent to be narrow, IBM argued that “you can’t look at this claim and say this claim is directed to just preserving state information because the meat of the claim is about how to do it. This claim is directed to recursively embedding state information in all continuations. That is what the invention is. The rest of the claim isn’t the invention. *The invention is the recursively embedding.*” D.I. 54 at 110:11-17. Yet, IBM now offers a construction of “recursively embedding” that reads “recursively” out of the claim. IBM repeats this tactic, for example, attempting to broaden the ‘967 and ‘849 Patent claims by using a dictionary definition of “network” unrelated to the specification to cover a scope to which the USPTO repeatedly confirmed it was not entitled.

## II. DISPUTED TERMS COMMON TO THE ‘967 AND ‘849 PATENTS

### A. Overview of the ‘967 and ‘849 Patents

The ‘967 and ‘849 Patents share a common specification that IBM contends discloses the basis for “the PRODIGY® Service.” Ex. A-5 at 1. The patents purport to improve upon existing “interactive computer networks” by (a) moving intelligence, including navigating between “applications,” to the user’s system, (b) transmitting, rather than page images, “objects” that can be combined to make “applications” for display, and (c) pre-fetching “advertising objects” so they can be later displayed without download delay. *See, e.g.*, ‘967 Pat. at 7:3-12, 10:50-64, 34:30-36. Like known “interactive computer networks,” the disclosed system aggregates content through a single central host computer that distributes content to user systems. *See id.* at 7:13-18



(“[n]etwork 10 thus eliminates the need to consult multiple information sources”).

IBM’s constructions seek to improperly stretch the claims to target a fundamentally different system—the world wide web—in which a user’s browser obtains content from countless separate web servers. IBM’s “network” construction would encompass a scope previously rejected by the USPTO in related applications. Further, the patents describe how to determine where one “application” ends and another begins only where the central host server maintains a list, providing no reasonable certainty on how to make that determination in other contexts. IBM’s other constructions also seek to improperly broaden the claims by ignoring the plain language and contradicting the specification, except for one term, “object,” that IBM irreconcilably argues should include some, but not other aspects of the preferred embodiment.

**B. objects**

<b>IBM:</b> separate data structures having a uniform, self-defining format that are known to the reception systems, including data types, such as interpretable programs and presentation data for display at the monitor screen of the user’s personal computer	<b>Defendants:</b> Plan and Ordinary Meaning
---	--

IBM seeks to substantially narrow the plain and ordinary meaning of “object” without any justification from the intrinsic record. First, the claim language itself demonstrates that when the inventors intended to narrow the term “object” beyond the plain meaning, they explicitly did so. For example, Claim 1 of the ‘967 Patent recites “data objects *having a prescribed data structure.*” ‘967 Pat. at 39:45; *see also* Ex. A-28 at 7 (“Applicants[] have amended their claims to expressly recite that the screen displays are generated ...from data objects having a prescribed data structure.”). Likewise, Claims 2 and 12 each recite “wherein *the data structure* of the objects *includes a header and one or more data segments.*” ‘967 Pat. at 39:62-63, 40:54-55.

Second, IBM’s construction comes from the “Description of the Preferred Embodiment” (*id.* at 4:53-54), particularly, a discussion of “RS 400” “in preferred form” (*id.* at 5:22):

RS 400 formulated in this fashion is capable of communication with the host system to receive information containing either of two types of data, namely objects and messages. Objects have a uniform, self-defining format known to RS 400, and include data types, such as interpretable programs and presentation data for display at monitor screen 414 of the user's personal computer 405.

'967 Pat. at 5:49-55. The language "RS 400 formulated in this fashion" confirms that this passage continues the discussion of RS 400 "in a preferred form," and nothing in the sentence indicates any intent beyond describing a preferred embodiment. *See id.* "It is not enough for a patentee to simply disclose a single embodiment or use a word in the same manner in all embodiments, the patentee must 'clearly express an intent' to redefine the term." *Thorner v. Sony Computer Entm't Am. LLC*, 669 F.3d 1362, 1365 (Fed. Cir. 2012) (quoting *Helmsderfer v. Bobrick Washroom Equip., Inc.*, 527 F.3d 1379, 1381 (Fed. Cir. 2008)).

Third, this passage is one of dozens describing properties of "objects" in the preferred embodiment, some broader and some narrower. IBM cannot pick and choose what it reads into the term. For example, other portions of the specification describe objects more narrowly, stating "objects ... are themselves made of further sub-blocks" and "are predefined, variable length records consisting of a fixed length header 551 and one or more self-defining record segments 552 a list of which is presented in FIG. 4c." '967 Pat. at 12:56-65.

Fourth, IBM's construction is contrary to IBM's statements to the USPTO. For example, in describing its invention, IBM described "separable units of data and program code; i.e., 'objects,'" without any of the limits it now seeks to inject into the claims. Ex. A-5 at 2. When IBM discussed various specification features of "objects" during prosecution, the USPTO criticized IBM for refusing to put concrete limits on "objects":

Attorney further argued that none of the applied references teach "objects" as defined in the specification and recited in dependent claim 2. Examiner asked Attorney to point out the definition of "objects" in the specification. ***Attorney stated that 'objects' comprise a 'header' and at least one 'segment' as described in the specification.*** Examiner asked Attorney to point out the definition of

“header” in the specification. *Attorney pointed to Fig. 4b, but stressed that the claims are not to be limited to the specific ‘headers’, ‘objects’, or ‘structures’ disclosed in the specification.* Therefore, Attorney is attempting to interpret the claim language in light of the specification for the purposes of overcoming the applied art, yet refuses to limit the claims as such.

Ex. B-5 at 2; *see also* Ex. B-19 at 5 (“During interviews, Examiner was unsuccessful in getting Applicant to commit to a definition for the word ‘objects.’”). In view of IBM’s failure to insert further limits in the claim language itself, the Board of Patent Appeals (“BPAI”) employed the same logic set forth in this Brief:

We do not read in the specifics of the data structure objects in the specification, Figs. 4a-4d, because appellants are capable of claiming the details of these data objects expressly, if desired, and because it is unknown how much of these structures should be included in an “object.”

Ex. B-3 at 8-9. The same conclusion follows here. Alternatively, should the Court apply IBM’s logic and read in specification properties, it should inject the entirety of the requirements of the preferred embodiment such as supporting each of the object types illustrated in Figure 4c.

### C. application(s)

<b>IBM:</b> Information events composed of a sequence of one or more pages opened at a screen	<b>Defendants:</b> Information events composed of a sequence of one or more pages opened at a screen to provide requested information and/or transaction operations. However, neither this construction nor the plain language provides sufficient guidance to a person of ordinary skill to determine what constitutes two different applications, rendering the claim indefinite.
---	---

The specification provides some guidance as to what is meant by “application:” (1) “Applications, *i.e.*, information events [] composed of a sequence of one or more pages opened at screen 414...,” and (2) “the user can begin to access various desired services (*i.e.*, partitioned applications) which provide display of requested information and/or transaction operations.” ‘967 Pat. at 9:33-34, 9:26-29. While this guidance allows formulation of a construction, neither that construction nor any other portion of the intrinsic record informs, with reasonable certainty, those skilled in the art of when one application begins and another ends. *See Nautilus, Inc. v.*

*Biosig Instr., Inc.*, 134 S.Ct. 2120, 2124 (2014); *see also Halliburton Energy Servs., Inc. v. M-I LLC*, 514 F.3d 1244, 1251 (Fed. Cir. 2008) (“Even if a claim term’s definition can be reduced to words, the claim is still indefinite if a person of ordinary skill in the art cannot translate the definition into meaningfully precise claim scope.”). That is, for any system other than the preferred embodiment, the ‘967 and ‘849 Patents fail to provide any objective basis to answer the question of what constitutes two different applications as opposed to two portions of the same application. Without that answer, one of ordinary skill cannot determine, with reasonable certainty, whether: (a) navigation is “between applications” (as opposed to within a single application) (‘967 Patent, Claim 1), (b) a “third screen partition” contains a “second application” (or just another part of a “first application”) (‘967 Patent, Claim 12), or (c) advertising is structured “in a manner compatible to that of the applications” (plural) (‘849 Patent, Claim 1).

And, this problem is only accentuated by IBM’s attempt to read these “interactive computer network” claims on modern web functionality. The patents describe a central entity “information source”—“network 10”—that provides clients access to “thousands” of applications, and enables the user to select among those applications. ‘967 Pat. at 20:38-44. In that system, the answer of what constitutes each application is simple: the central entity maintains a list applications. Specifically, network 10 maintains a catalog listing the applications and correlating each application to the “page template objects and object-ids of the entry screen (typically the first screen) for the respective application.” *Id.* at 18:54-60. Thus, simply consulting a list may be a sufficient answer for the “interactive computer network,” in which one central “information source” catalogs and identifies “applications.” But, this method is inapplicable to the disparate systems IBM accuses here. In this case IBM alleges that the “network” is the world wide web, and points to purported different “applications” within a single

website. There is no central entity that lists all “applications” on the world wide web, nor is “application” a label used to identify a specific grouping of content on the web.<sup>1</sup> Accordingly, if IBM’s infringement theory is accepted, a basis for distinguishing between applications other than a master list is needed, but the patents do not provide any.

The difficulty in understanding the boundaries between applications is also compounded by the specification’s failure to illustrate applications (plural); Figure 3b provides the only illustration of a screen from an actual application (singular):

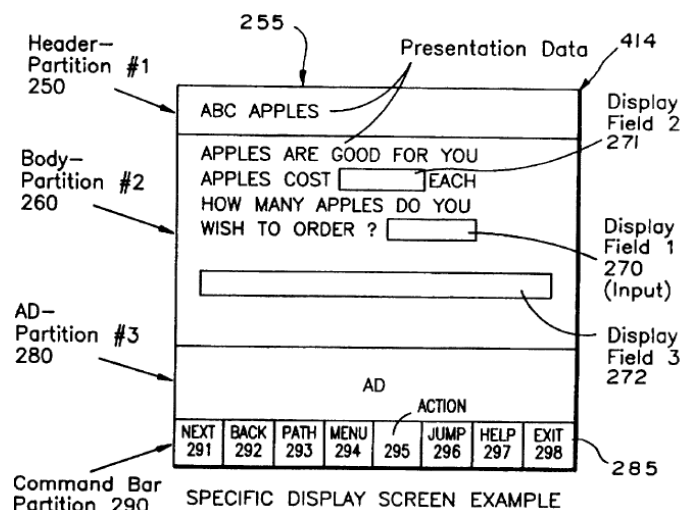


FIG. 3b

The patent describes Figure 3b as a screen display displaying the price of apples and calculating a total price when the user enters an amount to buy. *See id.* at 37:38-54. This sole display screen raises more questions than answers in how to distinguish between applications. In describing the sample application, the patents refer to it as the “apple application” (*id.* at 37:30), but elsewhere

<sup>1</sup> The patents also appear to use “application” interchangeably with a “page set” that is “built,” suggesting that something in the page programming indicates a grouping of pages to be considered an application, but provide no details on this concept. *Compare id.* at 17:42-49 (“NEXT command 291 causes the next page in **the current page set** to be built. If the last page of a page set has already been reached, NEXT command 291 is disabled by RS 400”) *with id.* at 3:23-24 (“the command bar features functions for progressing forward and backward **in the application**”). Again, this is a concept that cannot be applied to websites.

the patent refers to “the price of apples in a *grocery shopping* application” (*id.* at 29:42-45). The patent provides no guidance as to which of these labels is accurate. Likewise, the patent lists “*airline* reservations” (*id.* at 7:29-32), but during prosecution IBM provided a similar list of potential “applications,” this time including “*travel* reservations” (Ex. A-5 at 4).

Even if the sample application were limited to an “apple application,” the boundaries of that application are unclear. Does it include other pages? The patent states that this sample application “permit’s [sic] a user to purchase apples.” *Id.* at 36:47-48. But the displayed page does nothing but report a total price. This suggests that numerous other pages having other functionality such a shopping cart, forms for entering shipping and payment information, etc., are needed to facilitate the purchase of apples. Presumably these other pages would also be part of the same application, but the patent is silent on this issue. What if the application allowed the user to purchase apples for pick-up at a local store and then provided a page for generating driving directions from an entered location to the store location? Would that be a separate application or the same application? The patent does not say. As another example, what if the same vendor provided another page using the same template as Figure 3b, but pricing oranges instead of apples? Would that be a separate application or the same application?

Absent a central entity maintaining a master list of “applications,” the patent provides no way to answer these questions. Because the patents fail to inform, with reasonable certainty, those skilled in the art of when one application begins and another ends, the term “application” is indefinite. *See Nautilus, Inc.*, 134 S.Ct. at 2124; *see also Dow Chem. Co. v. Nova Chemicals Corp. (Canada)*, 803 F.3d 620, 635 (Fed. Cir. 2015) (claim invalid when multiple methods to determine claim scope with no guidance from intrinsic record regarding correct method).

**D. computer network / network**

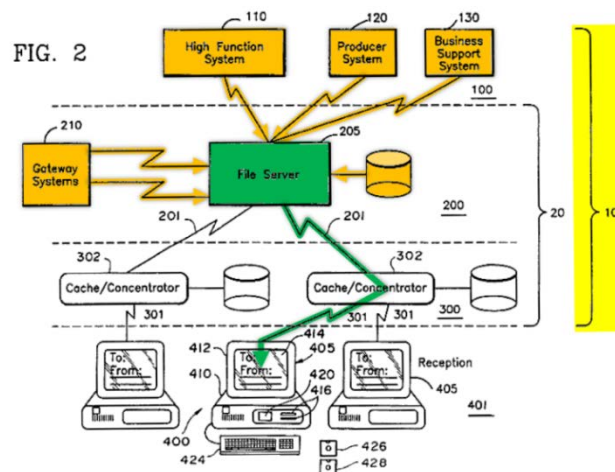
<b>IBM :</b> [no construction necessary or, alternatively:] two or more interconnected computers	<b>Defendants:</b> An information network consisting of a single central, host computer providing the objects to generate the screens of display at the reception systems
--	---

The term “network” has many potential “plain” meanings. While in some cases “network” may refer to an interconnection of computers, the term is also used to refer to a class of information services, such as in the phrase “broadcast network.” These patents use the term “computer network” to refer to a known class of online services provided by a central host computer. *See Phillips v. AWH Corp.*, 415 F.3d 1303, 1321 (Fed. Cir. 2005) (“the ‘ordinary meaning’ of a claim term is its meaning to the ordinary artisan after reading the entire patent”). IBM’s alternative construction takes a dictionary definition out of context, using the approach *Phillips* rejected of focusing “on the abstract meaning of words rather than on the meaning of claim terms within the context of the patent.” *Id.* IBM “is not entitled to a claim construction divorced from the context of the written description and prosecution history.” *Nystrom v. TREX Co.*, 424 F.3d 1136, 1144-45 (Fed. Cir. 2005) (citing *Phillips*, 415 F.3d at 1321). Indeed, the USPTO repeatedly rejected as unsupported IBM’s attempts in related applications to obtain claims covering architectures beyond a single central host computer providing all information to the reception system. *See* Ex. A-22 at 3, 7-8; Ex. A-25 at 6-7.

The patents consistently use the term “network” in accord with Defendants’ construction as describing a service. The patents explain that “[n]etwork 10 *provides information* on a wide variety of topics” and “network 10 *provides information, advertising and transaction processing services* for a large number of users simultaneously accessing the network.” ‘967 Pat. at 7:13-28. The patents also explain that “network 10 includes an improved procedure for searching and retrieving applications from the store of applications distributed throughout network 10.” *Id.* at 18:12-14. Thus, the patents consistently use “network” to refer to a service.

The patents not only use “network” to refer to a service, but also explain that the term is associated with providing that service using a particular architecture. They begin by explaining that “[i]nteractive computer networks are not new,” and “[t]raditionally they have included conventional, hierarchical architectures wherein *a central, host computer* responds to the information requests of multiple users.” *Id.* at 1:37-45. The patents go on to discuss purported improvements upon this “traditional” system as discussed in Section II.A, above, but the patents describe these improvements using the same the same “interactive computer network” architecture featuring a central host computer. *See, e.g., id.* at 7:54-63.

Indeed, the patent emphasizes the advantages of this single-source, central system, explaining that “[n]etwork 10 thus eliminates the need to consult multiple information sources” by “provid[ing] information on a wide variety of topics, including, but not limited to news, industry, financial needs, hobbies and cultural interests.” *Id.* at 7:13-18; *see also id.* at 7:34-37. The system architecture the patents disclose, primarily in Figure 2, is also consistent with Defendants’ construction. *See, e.g., id.* at 7:54-63. Specifically, file server 205 is the single central, host computer within network 10 providing the objects to generate the screens of display at the reception systems as shown in the annotated version of Figure 2, below:





File server 205 provides objects to the reception systems, either directly or through the use of cache/concentrator 302, which stores frequently-requested objects (and obtains objects it does not have from file server 205). *See id.* at 8:37-46. The patent states bluntly: “**All** active objects reside in file server 205.” *Id.* at 8:35. File server 205 obtains objects from various sources, including producer system 120. *Id.* at 8:35-38. Importantly, the patents acknowledge that “other sources of information and data; e.g., other computer systems” exist, but do **not** describe reception systems accessing such other computer systems. *Id.* at 5:13-18. Rather, “gateway systems 210” are “connected to server 205” as well as those “other sources” such that file server 205 obtains information “via the gateway 210.” *Id.*; *see also id.* at 7:58-63. Thus, Defendants’ construction is consistent with the usage of the term in the patents, and nothing in the patents entitles IBM to a broader construction.

This litigation is not the first time that IBM improperly attempted to extend this patent family beyond the disclosed and claimed “network.” During prosecution of several family members, the USPTO refused to allow claims directed to a reception system obtaining content from two different sources. In U.S. Patent Application No. 10/205,393, the USPTO held that the description of the “network architecture” (the same as in these patents) lacked written description support for obtaining advertising content from a second, different source. Ex. A-22 at 3, 7-8 (“Any request for content not already on the user system is fulfilled by the cache/concentrator 302 and file server 205.”); *see also* U.S. Patent Application No. 09/656,581, Ex. A-25 at 6-7 (same). Thus, adoption of IBM’s proposed construction would render the claims invalid for failing to comply with the written description requirement. *See Ruckus Wireless, Inc. v. Innovative Wireless Solutions, LLC*, Nos. 2015-1425 & 1438, 2016 WL 3065024, \*4 (Fed. Cir. May 31, 2016) (“The canon favoring constructions that preserve claim validity therefore

counsels against construing” the term in a way that “would likely render the claims invalid for lack of written description”).

### III. DISPUTED TERMS FROM THE ‘967 PATENT

A. the objects being retrieved from the objects stored at the respective reception system, or if unavailable from the objects stored at the respective reception system, then from the network

<p><b>IBM:</b> [no construction necessary or, alternatively:] the objects being retrieved from the objects stored at the respective reception system, or, if the current versions of the objects are not present from the objects stored at the respective reception system, then from the network</p>	<p><b>Defendants:</b> The objects stored at the respective reception system being retrieved from that respective reception system and, if not, retrieved from the network. That is, this language is not met if it is determined that an object is stored at the respective reception system, but it is retrieved from the network without first being retrieved from the respective reception system.</p>
--	--

Defendants’ construction (a) clarifies the plain language and (b) construes “unavailable,” each in a manner consistent with the intrinsic evidence. IBM’s construction introduces a “version” concept not found in the claim by incorrectly interpreting “unavailable.” Turning first to the plain claim language, the claim language presents a binary choice denoted by the “or if unavailable” bridging two actions: Is the object available from the local reception system? If yes, “retrieve[] from the objects stored at the respective reception system;” if no, retrieve “from the network.” The second sentence of Defendants’ construction confirms this binary choice.

Defendants’ construction is also consistent with the first step of a two-step logical flow the patent describes. ‘967 Pat. at 27:1-16, 30:66-31:13. Specifically, in the first step, “object storage facility determines whether the requested object *resides locally*, and if not, makes an attempt to obtain it from interactive network 10.” *Id.* at 27:3-9. For objects determined to reside locally, in the second step, “[o]bject storage facility 439 assures currency by requesting version verification from network 10 for those objects which are *available locally*....” *Id.* at 27:11-16. These two passages in tandem demonstrates that the patent uses the terms “available” and

“resides” interchangeably.

The patent also confirms that objects stored locally are retrieved during step one as a prerequisite to step two, explaining that object storage facility performs the currency check when “*an object stored at RS 400 is initially fetched* or accessed during a session.” *Id.* at 30:66-31:1. In the currency check, “a request to delivery system 20 is made for the object by specifying the version id *of the object stored at RS 400*,” which is stored in the object header. *Id.* at 31:1-3, 13:53-55. This retrieving is not optional as the patent does not disclose any way to perform the currency check other than obtaining the version id from the stored object header.

IBM’s construction is wrong because it incorrectly redefines “unavailable” to mean “the *current version*” is not stored. The claim phrase never mentions a current version concept, nor does the specification use “available” in this way. To the contrary, the patent explains that: “Object storage facility 439 assures currency by requesting version verification from network 10 for those objects which are *available* locally....” *Id.* at 27:11-16. That is, the system must determine whether an object is “available,” i.e., resides locally, before performing the currency check. Thus, whether an object is “available” or “unavailable” locally cannot hinge on a determination not yet made of whether the local copy is “the current version.”

**B. the method of claim 2 wherein the predetermined plan**

<b>IBM:</b> The method of claim 3 wherein the predetermined plan	<b>Defendants:</b> Indefinite
--	-------------------------------

Claim 4 recites “*the* predetermined plan,” which finds no antecedent basis in Claim 2. IBM recognizes this issue and seeks to rewrite the claim to refer to Claim 3. While Claim 3 recites “*a* predetermined plan,” there is no basis to change the dependency of Claim 4. Thus, IBM is correct inasmuch that Claim 4 contains a flaw, but that flaw is not correctable and, instead, renders Claim 4 indefinite. To obtain correction, IBM would need to show that “(1) the correction is not subject to reasonable debate based on consideration of the claim language and

the specification and (2) the prosecution history does not suggest a different interpretation of the claims. *Trusted Knight Corp. v. Int'l Bus. Machines Corp.*, 14-cv-1063-LPS-CJB, 2015 WL 7307134, at \*6 (D. Del. Nov. 19, 2015) (quoting *Novo Industries, L.P. v. Micro Molds Corp.*, 350 F.3d 1348, 1354 (Fed. Cir. 2003)). Neither is true here.

The original November 1994 versions of Claims 3 and 4 both plainly and logically depend from Claim 2. Ex. A-27 at 65. Each claim separately modifies the recitation “providing the first subgroup of command functions,” which appears in Claim 2:

2. The method of claim 1 wherein generating the second partition includes *providing the group of command functions with a subgroup of command functions*....

3. The *method of claim 2 wherein providing the subgroup of command functions* includes providing a command for causing the user to be presented with at least one procedures [sic] for navigating to a new application.

4. The *method of claim 2 wherein providing the subgroup of command functions* includes providing at least one command for causing the user to be presented with a plurality of different procedures for navigating to a new application.

*Id.* Indeed, issued Claims 5-8 employ this same style, each depending from Claim 4. ‘967 Pat. at 40:15-33. And, if Claim 4 was intended to depend from Claim 3, it would have said “wherein providing the subgroup of command functions *further* includes . . . .” It was not until, multiple amendments and years later, IBM added “predetermined plan” language to Claims 3 and 4 in its July 16, 1997, amendment that a problem arose:

3. (Thrice amended) The method of claim 2 wherein the objects are stored at the respective reception systems in accordance with a predetermined plan, and wherein providing the first subgroup of commands includes providing a command for causing the user to be presented with at least one procedure for navigating to a new application.

4. (Twice amended) The method of claim 2 wherein the predetermined plan for storing objects at the respective reception system includes providing the objects with a storage control parameter in their respective headers, and wherein providing the first subgroup of command functions includes providing at least one

command for causing the user to be presented with a plurality of different procedures for navigating to a new application.

Ex. A-3 at 2 (emphasis in original). This amendment created an inconsistency in the claim that cannot be solved.

IBM's proposed "correction" *is* subject to reasonable debate. Claim 4 contains two clauses, "providing the first subgroup," which undoubtedly refers to Claim 2, and "the predetermined plan." IBM's proposed "correction," while providing an antecedent basis for this second clause alters the original, intentional, and correct, reference to Claim 2 of the "providing the first subgroup" clause. Declining to alter this correct claim reference is, at a minimum, an alternative construction subject to reasonable debate. The prosecution history, in particular the fact that the original claims had no issues before IBM introduced the indefiniteness into the claims after several claim amendment, likewise does not support IBM's position. Accordingly, Claim 4 is indefinite because it is not "precise enough to afford clear notice of what is claimed, thereby appris[ing] the public of what is still open to them." *Nautilus, Inc.*, 134 S. Ct. at 2129.

**C. permit random movement**

<b>IBM:</b> allow navigation to new applications at the user's behest	<b>Defendants:</b> allow navigation to any of the computer network applications at the user's behest
---	--

Defendants' construction construes "random" consistent with the use of that term throughout the patent as allowing free movement to any of the applications available through the interactive computer network. It is unclear what, if any, meaning IBM assigns "random." Consistent with Defendants' construction, the patent equates "random" movement with the use of a series of search procedures: "The method further includes steps for providing command functions that facilitate *random* navigation to new applications with a variety of *different procedures....*" '967 Pat. at Abstract. These "procedures" are not a simple clickable link to a single new application, but are instead "for *searching* and retrieving applications from the store

of applications distributed throughout network 10.” *Id.* at 18:12-14. For example, the patent explains that the “JUMP” and “INDEX” commands “provide[] *random access to all services* represented by partitioned applications having key words.” *Id.* at 6:12-15. Use of these search procedures enables the user “[t]o select a particular partitioned application *from among thousands of such applications* residing either at the RS 400 or within delivery system 20 . . . .” *Id.* at 20:38-43.

Defendants’ construction is also consistent with Dependent Claims 5-8, which provide examples of “random” movement through the use of “procedures” including “randomly search the *available applications*,” “access an index of *available applications*,” “access a directory of application subject matter,” and “access a physical analogy of *available applications*.” *Id.* at 40:15-33. The specification description of these procedures is also in accord. *See id.* at 19:31-49 (invoking “Jump” to initiate “a subject matter *search of the network applications*,” invoking “Index” to initiate “an alphabetical *search of network applications*,” invoking “Guide” to initiate a series of graphic displays that present a “*physical description of the network applications*,” and invoking “Directory” to invoke a series of hierarchical menus presenting “*applications available on the network*”).

To the extent IBM’s construction is intended to encompass a single link with a single destination application/page, the patent never uses “random” in such a context, nor do any of the “procedures” the patent describes as enabling “random” movement constitute such a link.

**D. at least one procedure for navigating to a new application / a plurality of different procedure [sic] for navigating to a new application**

<b>IBM:</b> [no construction necessary or, alternatively] at least one procedure for moving to a new application / a plurality of different procedures for navigating to a new application	<b>Defendants:</b> Procedure(s) enabling the user to search and select among any application in the computer network.
--	---

Defendants’ construction equates “navigating” with “searching and selecting,” as the

patent consistently uses the term. The patent begins its discussion of the “*navigation* interface” by emphasizing that “network 10 includes an improved procedure for *searching* and retrieving applications from the store of applications distributed throughout the network.” ‘967 Pat. at 18:11-14. Specifically, network 10 purportedly improves upon prior interactive network navigation schemes by preparing “a library of tables” of “the active applications available on network 10” to support “a plurality of operations which appear at the user’s screen as different search techniques.” *Id.* at 18:50-19:19. The patent then goes on to describe specific operations— “jump,” “index,” “directory,” and “guide” also claimed in Claims 5-8 as examples of the procedures for navigating. *See id.* at 19:20-56, 40:17-37. Thus, “navigating” refers to the process of identifying a new application, i.e., “searching,” not just “moving” after identification.

The procedures for searching must be for any application in the computer network for multiple reasons. As discussed in Section III.C, above, the patent describes each of the exemplary procedures both in the specification and Dependent Claims 5-8 as searching “*applications available on the network.*” *See* ‘967 Pat. at 19:31-49, 40:15-33. More generally, Claims 3 and 4 in which these terms appear each depend from Claim 2, which itself requires navigation to all available applications as discussed in Section III.C, above. Further, during prosecution IBM specifically argued that “indexed or directory *navigation of applications* or the use of predetermined progression of transitions; i.e., ‘path’, through applications” was “fundamentally different” than selecting between applications in Microsoft Windows because “Applicants’ approach is that of a structured, database-type navigation *through a potentially, large number of unseen applications.*” Ex. A-4 at 8. Thus, navigating to a “new application” refers to searching for that new application among any application in the computer network.

**E. storage control parameter**

<b>IBM:</b> a parameter that identifies the storage characteristic for the object	<b>Defendants:</b> a parameter that identifies the storage characteristic for the object for initial and/or continued storage
---	---

The patent describes an “object storage facility” making two types of storage decisions—whether to initially store an object and whether to continue to store an object. *See* ‘967 Pat. at 27:1-16, 30:66-31:13. Defendants’ construction clarifies that parameters used in not only the initial, but also the continued storage decision are “storage control parameters,” consistent with the patent, while IBM’s construction is vague on that issue. The patent explicitly characterizes parameters used in this second decision as “storage control parameters.” Specifically, the patent explains: “The currency of objects stored locally at the RS 400 is verified before use according to the object’s *storage control parameters* and the storage criterion in use for *version checking*.” *Id.* at 6:60-64. Thus, Defendants’ construction reflects the proper scope.

**IV. DISPUTED TERMS FROM THE ‘849 PATENT****A. storing a predetermined amount of the advertising data in a store established at the respective reception systems**

<b>IBM:</b> [no construction necessary or, alternatively:] storing a certain amount of advertising data in a store established at the respective reception systems	<b>Defendants:</b> Storing a certain (i.e., specific) amount of advertising data determined independent of the requested applications in a store established at the respective reception systems
--	--

IBM’s construction fails to give meaning to the word “predetermined,” improperly rendering it superfluous. *See Merck & Co. v. Teva Pharm. USA, Inc.*, 395 F.3d 1364, 1372 (Fed. Cir. 2005) (“A claim construction that gives meaning to all the terms of the claim is preferred over one that does not do so.”); *TQ Beta LLC v. Dish Network Corp.*, 14-CV-848-LPS-CJB, 2016 WL 356024, at \*5 (D. Del. Jan. 28, 2016) (same). Defendants’ construction makes clear that “predetermined” means “independent of the requested applications” because that amount must be determined before and independently of a requesting application, consistent with the



patent and IBM's arguments to the USPTO. Specifically, the patent discloses "ad manager 442" "pre-fetching" advertising objects "so they are available in RS local store 440 *when requested by object interpreter 435 as it builds a page.*" '849 Pat. at 33: 63-65. Ad manager knows when more objects must be "pre-fetched" when "the number of remaining advertising objects[] falls below *the pre-fetch advertising threshold.*" *Id.* at 33:65-34:3; *see also id.* at 33:55-58 (preferred value of "pre-fetched advertising object threshold of 3"). Thus, this "pre-fetch advertising threshold" sets forth a predetermined amount that is determined independently of any specific requested application. That is exactly what IBM said this language meant during prosecution: "The specification also points out how *the amount of data which is pre-fetched* (55:16-20) is established, *thus providing support for the 'predetermined amount of the advertising data.'*" Ex. B-18 at 27. This parameter is set "by the logon application program object," confirming it is independent of the number of advertisements included on any given page. '849 Pat. at 33:48-54.

Moreover, during the prosecution history of the '849 Patent, the BPAI granted IBM's request for reconsideration acknowledging that because a given videotext page could include any number of advertisements, the amount of advertising stored as a result of storing such a page cannot be a "predetermined amount." Ex. B-11 at 42 ("There appears to be no way the amount of advertising in the page store can be considered to be 'predetermined' because videotext pages do not have to have advertising or advertising in a fixed amount."). Thus, Defendants' construction equates "predetermined" with "independent of the requested applications," as IBM successfully argued to the USPTO, and consistent with the specification.

**B. selectively storing advertising objects at a store established at the reception system**

<b>IBM:</b> storing advertisement objects if they meet certain criteria, such as being non-volatile, non-critical to network integrity, or if they are critical to ensuring reasonable response time, at a store established at the reception system	<b>Defendants:</b> pre-fetching advertising objects and storing at a store established at the reception system in anticipation of display concurrently with the applications
--	--

Defendants’ construction interprets “selectively storing” consistent with the patent’s disclosure of selecting *advertising* objects for storage. In contrast, IBM’s construction tracks a preferred embodiment of storing objects *other than advertising objects* for storage and is, in fact, contrary to the preferred embodiment of the patent for selecting *advertising* objects for storage. Specifically, the patent explains “the user reception system ... includes facility for *storing* and managing the *advertising* so that it can be *pre-fetched* from the network and staged at the reception system *in anticipation of being called for presentation*.” ‘849 Pat. at 3:16-21. Because the advertising objects are “pre-fetched” for later display, all fetched advertising objects must necessarily be stored in anticipation of later display. Thus, “selectively” refers to the selection of specific advertising objects to pre-fetch, all of which are then stored. This is exactly what IBM explained to the USPTO this language means:

[T]he specification describes **selectively storing advertising objects at the user reception system** so that when advertising is to be presented, its data might be found available at the reception system without going back to the host (6:7-12; 14:32-15:23). The method which is described provides for storing and managing advertising objects so that **advertising objects may be separately prefetched from the network** and cached at the reception system in anticipation of being called for presentation.

Ex. B-8 at 6-7 (emphasis in original).

Defendants’ construction is also consistent with Dependent Claims 2-7, which recite various additional details of the pre-fetching process. For example, Claim 3 recites “wherein storing advertising objects . . . includes storing advertising object identifications.” ‘849 Pat. at

39:66-40:1. This is exactly how pre-fetching is performed: “Ad manager 442 pre-fetches advertising objects by passing *advertising object id’s* from the advertising queue . . . .” *Id.* at 33:59-63. Claim 3 further recites that the object identifications are based on a characterization of the users (*id.* at 40:1-4), again consistent with pre-fetching using “object id’s targeted to the specific user.” *Id.* at 33:17-27. Defendants’ construction is also consistent with the file history, in which the BPAI confirmed that “selectively” refers to selecting of advertising objects. *See* Ex. B-3 at 9 (“[W]e interpret ‘selectively storing’ to mean that advertising objects (entities) are selected by the system for storing at the reception system, as opposed to being part of videotex page requested by the user.”).

IBM’s construction appears to be based on the patent’s description of use of a storage candidacy value by object storage facility 439 in selecting objects for storage, which is inapposite to selection of *advertising* objects for storage. *See, e.g., id.* at 27:47-28:18. The patent explains that objects with one storage candidacy value, i.e., “trashable,” will not be stored separate from immediate display, while objects with another values such as “cacheable” can be retained over a session. *Id.* But, the patent never suggests a storage candidacy value be used to selectively store (i.e., store some but not others) *advertising* objects. To decline to store any pre-fetched advertising objects would be nonsensical because the pre-fetched objects are not immediately displayed; they must be stored for later display.

**C. structuring advertising in a manner compatible to that of the applications so that it may be presented**

<b>IBM:</b> [no construction necessary or, alternatively:] organizing advertising in a manner compatible to that of the applications so that it may be presented	<b>Defendants:</b> Formatting advertising for use with a plurality of applications
--	--

Both parties’ constructions confirm that this language requires that an advertisement be “structured” in a way that it may be used with applications (plural). During prosecution IBM

distinguished this language from an advertisement arranged within a single application:

[T]here is no suggestion (much less any description) that **plural** applications are structured (“structuring applications”) as well as structuring advertising to be compatible, **not only with one application, but to be compatible with many applications** (“structuring advertising in a manner compatible to that of the applications”).

Ex. B-4 at 11 (emphasis in original). The two constructions otherwise differ in whether “structuring” is replaced with “organizing” or “formatting.” Defendants’ proposal, “formatting,” is taken from the BPAI’s decision analyzing this term. *See* Ex. B-2 at 37-38. It is unclear what distinction IBM is attempting to make by using a different term.

## V. DISPUTED TERMS FROM THE ‘601 PATENT

### A. Overview of the ‘601 Patent

The ‘601 Patent generally relates to “preserving state in computers communicating over networks, such as the WWW using stateless protocols, e.g., HTTP.” ‘601 Pat. at Abstract. The patent does not purport to have been the first to identify the need to maintain state or to implement solutions. *See id.* at 7:33-9:37 (background section discussing “Current Methods for Handling State on the Web”). Rather, the patent discloses, and the Asserted Claims claim, a specific retrofitting implementation in which a specialized “converter program” performs embedding of state information separate and apart from the dynamic web page generation process. *See id.*; *see also* D.I. 54 at 110:11-13 (IBM arguing “you can’t look at this claim and say this claim is directed to just preserving state information because the meat of the claim is about how to do it.”). Specifically, the “converter program” intercepts web pages generated in response to a client request before being sent back to the client: “Instead of returning the output, page `h` to the client unmodified,” the completely generated page is “pass[ed] to a convert1 module of the converter,” which “modifies all the hypertext links” “to preserve the state variables.” ‘601 Pat. at 14:6-64. Each of the asserted independent claims further breaks down this process of modifying

all links (“continuations” in the claims) into steps of, first, “identifying all continuations” in the received output of the page generation process and, second, “recursively embedding the state information in all identified continuations.” *See, e.g., id.* at 23:56-58.

IBM’s proposed constructions, including refusing to acknowledge the order of steps dictated by the claim language, reading “recursively” out of “recursively embedding,” and reading “predetermined” out of “predetermined criteria” in the “filtering” and “adding” steps, seek to escape the limits of the plain language. These attempt should be rejected. *See also* D.I. 54 at 110:13-17 (IBM arguing “[t]his claim is directed to recursively embedding state information in all continuations. That is what the invention is. The rest of the claim isn't the invention. The invention is the recursively embedding.”).

**B. order of the independent claim steps**

“A claim ‘requires an ordering of steps when the claim language, as a matter of logic or grammar, requires that the steps be performed in the order written, or the specification directly or implicitly requires’ an order of steps.” *Mformation Techs., Inc. v. Research in Motion Ltd.*, 764 F.3d 1392, 1398 (Fed. Cir. 2014) (quoting *TALtech Ltd. v. Esquel Apparel, Inc.*, 279 F. App’x. 974, 978 (Fed. Cir. 2008)).<sup>2</sup> Such is the case here where the plain language of each of the four asserted independent claims dictates that the claimed steps/functions be performed in order. First, the plain language of Claim 14 mandates that the “detecting” through “communicating” steps occur in order. The claim recites, first, detecting a “request for *a service*” and, second, “performing *said service*,” with the antecedent basis of “service” confirming the first “detecting” step must occur before the second “performing” step. ‘601 Pat. at 19:36-38. The third step,

---

<sup>2</sup> IBM has declined to take a position as to whether it agrees or disagrees with the required step order, stating only that the plain language is enough. However, “[t]here is nothing improper about considering the order of steps issue in connection with claim construction.” *Cronos Techs., LLC v. Expedia, Inc.*, No. 13-1538-LPS, slip op. at 20 (June 20, 2015).

“identifying all continuations in *an output from said service*” (*id.* at 19:38-39) must occur after the second step of “performing” in order for an “output” to exist (and the claim explicitly states performing and identifying are “in response to said step of detecting” (*id.* at 19:39-40)). The fourth step of “recursively embedding the state information in all *identified continuation*” (*id.* at 19:41-42) must occur after the third “identifying” step, which creates the input to this step. Finally, the claim explicitly states that the fifth step of “communicating” is “in response to said step of embedding.” *Id.* at 19:43-44.

Next, Claim 40 recites means-plus-function elements with identical language to Claim 14 for “detecting,” “identifying,” and “recursively embedding” functions, and those functions must be in the same order for the same reasons. *Id.* at 22:46-51. As to the function of “communicating *the output* to the client,” as discussed in detail below, the only structure clearly linked to this function includes an algorithm for communicating after embedding. *See* Section V.E, below.

Finally, Claims 51 and 60 contain nearly identical language to each other, each beginning with a first step of “receiving a service request.” ‘601 Pat. at 23:54-55, 24:43-44. The second step of “identifying all continuations in *an output from said service*” (*id.* at 23:56-57, 24:45-46) must follow the first step of “receiving a service request” since prior to receipt of the request to perform the service there is no performance or output of the performance. The third step of “recursively embedding the state information in all *identified continuations*” (*id.* at 23:57-58, 24:46-47) must occur after the second “identifying” step, which creates the input to this step. Finally, the fourth step of “communicating a response including the continuations and *embedded state* information” (*id.* at 23:60-61, 24:49-50) must follow the third “recursively embedding” step as there is no “embedded” state information to communicate prior to the “recursively embedding” step.

**C. recursively embedding the state information in all identified continuations**

<b>IBM:</b> modifying each identified continuation to include state information	<b>Defendants:</b> repeatedly applying a program to each identified continuations to modify all identified continuation to include state information
---	--

IBM's construction fails to give meaning to the word "recursively," improperly rendering it superfluous. *See Merck & Co.*, 395 F.3d at 1372. That is, IBM defines "embedding the state information in all identified continuations." IBM's failure to give meaning to this term is particularly egregious in view of IBM's repeated pronouncements that "[t]he invention is the recursively embedding." D.I. 54 at 110:16-17. Defendants' construction is consistent with the specification and known understanding of a "recursive" function in computer science of a program that repeatedly runs. For example, the IBM Dictionary of Computer, Tenth Edition (Aug 1993) explains that the term "recursive" pertains to a program "that calls itself after each run until it is interrupted or until a specified condition is met." Ex. C-4; *see also* Exs. C-5 ("recursion" definition), C-6 ("recursive routine" definition), C-7 ("recursion" definition). Thus, the plain language requires applying the same program repeatedly until a condition (i.e., no more identified continuations are left to modify) is met.

Consistent with the plain language, the patent discloses a program, "convert1," that "modifies all the hypertext links to HTML in [the received output file], to preserve the state variables" (step 811) and "modifies all hypertext links to CGI programs" to embed state information (step 812).<sup>3</sup> '601 Pat. at 14:6-64. Specifically, convert1 receives the output, "an HTML page `h` with hypertext links," generated by a CGI program, "P1," and modifies all links to HTML files and the links to CGI files within the output file h. *See id.* For each of steps 811

<sup>3</sup> The patent discloses that in the approach of Fig 9b, convert1 does *not* "embed the state variables within any hypertext links generated by the CGI program," in contrast to the preferred approach of Fig. 9c in which convert1 does "embed the state variables . . . within hypertext links generated by the CGI program." '601 Pat. at 14:47-62. The approach of Fig. 9b is an unclaimed approach as it is contrary to the plain claim language.

and 812 the patent describes how to convert an individual link. *See id.* Confirming that convert1, consistent with the term “recursively,” operates by systematically passing through each encountered continuation and performing the disclosed conversion on that continuation, the patent explains that “the modification of links to HTML files (step 811) and links to CGI programs (step 812) could be done in a *one-pass* or a *two-pass* process.” *Id.* at 14:26-29. Thus, Defendants’ construction is consistent with the plain meaning of the claim, the preferred embodiment, and IBM’s arguments.

**D. continuation**

<b>IBM:</b> a new request in a conversation which a client may send to a server, such as, for example, a hyperlink	<b>Defendants:</b> code within an output that, upon selection by the client, causes the client to send a new request to a server, such as, for example, a hyperlink
--	---

Both parties take their proposed constructions from an explicit definition in the specification, and both describe a relationship to other claim terms, but IBM’s construction adds ambiguity and raises multiple Section 112 issues. Both constructions start from:

Continuations: Hypertext links (or hyperlinks) are examples of continuations in client-server communications. A continuation is a new request which a client may send to a server.

‘601 Pat. at 2:48-50. Defining a continuation as “a new request” is not sufficient because the client could also make a new request by manually entering a URL, but that is not a continuation. *See id.* at 2:61-64, 7:9-11 (“A conversation is interrupted whenever the client obtains a new page by explicitly requesting a new URL *instead* of following hypertext links.”). Defendants’ construction thus adds language to the front of the definition to provide the proper context.

To the extent IBM’s construction addresses the same issue, it does so in a way that is nonsensical and adds ambiguity. The patent explains the relationship between a “continuation” and “conversation”: “a client engages in a conversation whenever it follows hypertext links,” i.e., continuations, in the received response from the server, and “[a] conversation is interrupted



whenever the client obtains a new page by explicitly requesting a new URL instead of following hypertext links.” *Id.* at 2:61-64, 7:9-11. Thus, a “continuation” exists when the page is created and only by later user action does a continuation become part of a conversation. That is, there is no way to determine whether a continuation is “in a conversation” when a page is created.

Of more concern, IBM’s vague construction appears to be an attempt to preserve the ability to assert infringement even where state information is not recursively embedded in *all identified continuations*, as each independent claim requires. That is, IBM should not be permitted to argue that a hyperlink in a web page is not a continuation because it is not “in a conversation.” Such an argument would not only be contrary to the plain claim language, but such a claim scope would render the claim invalid for lack of written description and enablement. The patent includes no disclosure on how to “identify[] *all* continuations in an output” if “*all* continuations” is something less than “*all* the hypertext links.” *See, e.g., id.* at 14:15-17.

**E. means-plus function terms**

The parties agree that four terms (all of which appear in Claim 40) are subject to Section 112(f), have agreed functions tracking the plain language, and have clearly linked disclosure in the form of the server performing disclosed algorithms. IBM also asserts corresponding structure *on the client*. But the specification’s brief, aspirational, and non-enabling discussion of moving unspecified portions of functionality from the server to the client is insufficient to disclose and “clearly link” algorithms for performing the claimed functions on the client. “The duty of a patentee to clearly link or associate structure with the claimed function is the quid pro quo for allowing the patentee to express the claim in terms of function under section 112, paragraph 6.” *Med. Instrumentation & Diagnostics Corp. v. Elekta AB*, 344 F.3d 1205, 1211 (Fed. Cir. 2003).

Focusing first on the functions of “identifying all continuations” and “recursively

embedding” state information in the identified continuations, the parties agree that the server performing certain aspects of “the convert1 module of the converter program” shown in Figures 4, 7a, and 8, constitutes structure clearly linked to the claimed functions. In arguing the client performing these same algorithms is also corresponding structure, IBM cites to the following paragraph that does not even mention “convert1” specifically:

Another application of downloadable server code to the present invention would be to allow the ‘converter’ 416 to run on the client. Here, clients would download all or part of the ‘converter’ logic 416 from the server 410’ to the client for execution. This would allow the full functionality of the present invention with all (or part of) the processing taking place locally on the client 450.

‘601 Pat. at 16:30-36. This passage bears a tangential, at best, relationship to the certain aspects of “the convert1 module of the converter program” that the parties agree constitutes a disclosed algorithm. *See id.* To the extent IBM argues that the specification links the convert1 functionality by referencing “all or part of the ‘converter’ logic,” this vague reference falls short of “clearly linking” client-side structure to the claimed functions, the quid pro quo of Section 112(f) claiming. *Med. Instrumentation & Diagnostics Corp.*, 344 F.3d at 1211.

Regardless, the vague suggestion of moving converter logic to the client is insufficient to disclose “clearly linked” structure because that language does not include sufficient teachings for one of ordinary skill to develop an actual functional implementation that could be corresponding structure. The patent explains that three server-based components—a CGI program generating the pages, the “convert1” module, and the “convert2” module—must pass information to each other *within the server* to function. ‘601 Pat. at 14:8-11, 15:49-52, 15:61-67. Specifically, convert1 cannot perform its recursively embedding function (including the preliminary step of identifying all continuations in which to embed) until either the CGI program (*id.* at 14:8-11) or convert2 (*id.* at 15:49-52) passes it the state information to embed. Likewise, the CGI program cannot generate a new page based on the client request and state information until convert2

extracts that state information from the request and passes it. *Id.* at 15:61-67. Thus, the bare suggestion to move convert1 and/or convert2 to the client, while the CGI program remains on the server, would sever the required internal communication links between the CGI program and the module resulting in a non-functional system.

Likewise, the server-side algorithmic steps the parties agree correspond to “detecting when the request for a service requires preservation of the state information,” steps 510 and 810 are not readily transferred to the client side. For example, block 510 of Figure 4 takes as its input a client request received at the server and determines whether to route that request to the CGI program or converter module. This functionality cannot simply be moved to the client side without further implementation details, which the patent does not include.

Finally, for “communicating the output to the client,” this again is a server function with a single disclosed corresponding algorithm: “The modified output is then returned to the requesting client.” *Id.* at 15:33-34. IBM again cites to 16:30-43, but this section never even mentions communicating an *output*; it only mentions computer “code.” Thus, it discloses nothing on the client “clearly linked” to the claimed function.

**F. [dynamically downloading computer program code to the client to] perform said step of embedding which is responsive to said step of communicating the output to the client**

<b>IBM:</b> [no construction necessary]	<b>Defendants:</b> Indefinite
---	-------------------------------

This phrase, which appears in Dependent Claim 63, directly contradicts Independent Claim 60, from which it depends, rendering Claim 63 indefinite. Specifically, and as discussed above in Section V.B, Claim 60 recites “recursively embedding the state information...” and “communicating a response including the continuations and embedded state information.” ‘601 Pat. at 24:46-50. This “communicating” step necessarily follows the “embedding” step because the “embedded information” cannot be communicated until it exists (i.e., the embedding is

performed). Yet, Claim 63 attempts to reverse this order, stating that “*said* step of embedding” “is responsive to,” meaning it is later in time, than “*said* step of communicating.” *Id.* at 25:4-7. The requirements of Claims 60 and 63 are thus incompatible—one or the other must be ignored, and they cannot both be true. Accordingly, one of ordinary skill would be unable to determine, with reasonable certainty, the scope of Claim 63. *See Nautilus, Inc.*, 134 S.Ct. at 2124.

**G. filtering one of said hyperlinks and data output from said services according to a predetermined criteria / adding one of said hyperlinks and data to said output from said services according to a predetermined criteria**

<b>IBM:</b> removing one of said hyperlinks and data output from said services according to criteria determined prior to removing	<b>Defendants:</b> Removing from said output of said services one of said hyperlinks and data according to criteria determined prior to receiving said request from said client.
<b>IBM:</b> inserting one of said hyperlinks and data to said output from said services according to criteria determined prior to inserting	<b>Defendants:</b> Inserting in said output of said services one of said hyperlinks and data according to criteria determined prior to receiving said request from said client.

The parties’ constructions are largely the same and both recognize the need to address the phrase “*predetermined* criteria” and answer the question “determined prior to what?” IBM provides a circular construction that reads “predetermined” entirely out of the claim, while Defendants’ construction answers the question consistent with the specification. IBM proposes determining the criteria for “*removing*” “prior to *removing*” and the criteria for the “*inserting*” “prior to *inserting*.” These are non-limits. The criteria for performing an operation are necessarily determined by the instant the operation is performed and, thus, IBM’s construction encompasses determining criteria at any possible time, improperly rendering “predetermined” superfluous. *See Merck & Co.*, 395 F.3d at 1372.

In contrast, Defendants’ constructions are consistent with the specification examples of filtering and removing. The patent provides two primary examples. First, “filter[ing] all HTML text and leav[ing] out phrases and hypertext links which have been determined to be

objectionable.” ‘601 Pat. at 17:18-19. Here, the predetermined criteria is whatever has been “determined to be objectionable.” *See id.* Second, establishing a “a database 475 of home page URL’s for major corporations” and “add[ing] hypertext links each time the name of a company in the database appears in an HTML page.” *Id.* at 17:36-43. Here, the criteria is whether a company name is in a database. *See id.* In each case, the criteria is unrelated to any particular user request and, thus, may be determined before such a request is made.

## VI. DISPUTED TERMS FROM THE ‘346 PATENT

### A. federated computing environment

<p><b>IBM:</b> a set of distinct entities, such as enterprises, organizations, institutions, etc., that cooperate to provide a single-sign-on, ease-of-use experience to a user, wherein the enterprises need not have a direct, pre-established, relationship defining how and what information to transfer about a user</p>	<p><b>Defendants:</b> A set of distinct entities, such as enterprises, organizations, institutions, etc., that cooperate to provide a single-sign-on, ease-of-use experience to a user. A federated environment differs from a typical single-sign-on environment in that two enterprises need not have a direct, pre-established, relationship defining how and what information to transfer about a user.</p>
---	---

Both parties take their proposed constructions from an explicit definition in the specification, but IBM truncates the language in a way that makes the remaining language misleading. Specifically, the complete language explains that the federated computing environment “*differs* from a typical single-sign-on environment in that two enterprises *need not* have” a relationship with certain characteristics. ‘346 Pat. at 10:65-11:1. In context, this full sentence sets forth a negative limitation explaining how the claimed system “differs” from the “typical” prior art. *Id.* That is, in this sentence, “need not” means “does not.” IBM’s truncated version improperly changes the context.

## VII. CONCLUSION

Defendants’ constructions should be adopted for the reasons above.

Dated: July 26, 2016

*Of Counsel:*

Dan D. Davison  
NORTON ROSE FULBRIGHT US LLP  
2200 Ross Ave., Suite 3600  
Dallas, TX 75201  
(214) 855-8000  
dan.davison@nortonrosefulbright.com

Richard S. Zembek  
Daniel S. Leventhal  
Daniel A. Prati  
NORTON ROSE FULBRIGHT US LLP  
Fulbright Tower  
1301 McKinney, Suite 5100  
Houston, TX 77010  
(713) 651-5283  
richard.zembek@nortonrosefulbright.com  
daniel.leventhal@nortonrosefulbright.com  
dan.prati@nortonrosefulbright.com

Gilbert A. Greene  
Marwan Elrakabawy  
NORTON ROSE FULBRIGHT US LLP  
98 San Jacinto Boulevard, Suite 1100  
Austin, TX 78701  
(512) 474-5201  
bert.greene@nortonrosefulbright.com  
marwan.elrakabawy@nortonrosefulbright.com

/s/ Francis DiGiovanni

Francis DiGiovanni (#3189)  
Thatcher A. Rahmeier (#5222)  
DRINKER BIDDLE & REATH LLP  
222 Delaware Avenue, Suite 1410  
Wilmington, DE 19801  
(302) 467-4200  
francis.digiovanni@dbr.com  
thatcher.rahmeier@dbr.com

*Attorneys for Defendants The Priceline  
Group Inc., Priceline.com LLC, KAYAK  
Software Corporation, and OpenTable, Inc.*